

**Система хранения и предоставления электронных документов
«БФТ.е-Архив»**

Руководство программиста

на 27 листах

Аннотация

Настоящий документ содержит основные положения и сведения, необходимые для работы программиста с Системой хранения и предоставления электронных документов «БФТ.е-Архив».

Документ разработан согласно требованиям следующих нормативных документов: ГОСТ Р 59795–2021 «Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов», ГОСТ Р 59853–2021 «Информационные технологии (ИТ). Комплекс стандартов на автоматизированные системы. Термины и определения».

Содержание

Аннотация	2
Список принятых терминов и сокращений.....	5
1 Общие положения	7
1.1 Наименование программы.....	7
2 Назначение и условия применения программы	8
2.1 Назначение программы.....	8
2.2 Функции, выполняемые программой	8
2.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы.....	8
2.3.1 Требования к составу и параметрам комплекса технических средств.....	8
2.3.2 Требования к общесистемному программному обеспечению	9
2.3.3 Требования к персоналу	12
3 Характеристика программных средств программы.....	13
3.1 Описание основных характеристик программы	13
3.1.1 Временные характеристики программы	13
3.1.2 Режим работы программы	13
3.1.3 Средства контроля правильности выполнения программы	14
3.1.4 Средства контроля самовосстанавливаемости программы.....	14
4 Обращение к программе	15
4.1 Запуск процедур для интеграции с внешними системами	15
4.2 Создание скриптов	15
4.2.1 Создание скриптов для планирования заданий в программе	15
4.2.2 Примеры скриптов в Kotlin для Планировщика заданий	17
4.2.3 Используемые регулярные выражения в скрипте.....	17
5 Входные и выходные данные.....	19
5.1 Характер и организация входных данных	19
5.1.1 Описание входных данных, получаемых с использованием API.....	19
5.2 Характер и организация выходных данных	19
6 Сообщения	20
Приложение А Описание программного интерфейса приложения (API).....	21
A.1 Получение токена логина	21
A.2 Сохранение документа.....	21
A.3 Сохранение вложений документа.....	23
A.4 Сохранение подписей для вложения документа	24

A.5	Получение документа	24
A.6	Получение вложения.....	25
A.7	Получение подписи вложения	25
A.8	Удаление вложений из последней версии документа	26
A.9	Удаление подписей вложения.....	26

Список принятых терминов и сокращений

Термин / Сокращение	Определение термина / Расшифровка сокращения
API	(англ. Application Programming Interface) – прикладной программный интерфейс системы
Kotlin	Язык программирования
REST	(англ. Representational State Transfer) – архитектурный стиль взаимодействия компонентов распределённого приложения в сети
SQL	(англ. Structured Query Language, «язык структурированных запросов») – декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных
JSON	(англ. JavaScript Object Notation) – текстовый формат обмена данными, основанный на JavaScript
XML	(англ. Extensible Markup Language) – расширяемый язык разметки
XSD	(англ. XML Schema definition) – язык описания структуры XML-документа
БД	База данных – совокупность взаимосвязанных данных, организованных в соответствии со схемой базы данных таким образом, чтобы с ними мог работать пользователь
Вложение	Файл, присоединенный к ЭД, почтовому сообщению, справочнику Системы
Документ	Сущность информационной системы, логически являющаяся документом информационной системы
Карточка	Объект БФТ.е-Архив, содержащий общую информацию об ЭД, информацию о связанных с ЭД Вложениях и ЭП Вложений, информацию о вложениях почтовых сообщений, справочников Системы
Карточка Документа	Объект Системы, содержащий атрибуты Документа. Предназначен для поиска вложений по атрибутам Документа в Карточке Системы. Состоит из атрибутов Документа, Вложений Документа, ЭП Вложений Документа (при их наличии), истории изменений Карточки Документа (версий)
Карточка	Реестр Карточек Документов и Реестр Карточек Объектов. Имеет иерархическое представление по учетным системам, подключенным к Системе, и типам документов учетных систем
Карточка объекта	Объект Системы, содержащий атрибуты Объекта. Предназначен для поиска вложений по атрибутам Объекта в Карточке Системы. Состоит из атрибутов Объекта, Вложений Объекта, ЭП Вложений Объекта (при их наличии), истории изменений Карточки Объекта (версий)
Объект	Сущность информационной системы, логически не являющаяся документом информационной системы (почтовое сообщение, справочник, медиа-файл и т.д.)
ОЗУ	Оперативное запоминающее устройство – техническое устройство, реализующее функции оперативной памяти
Система «БФТ.е-Архив», Система	Система хранения и предоставления электронных документов «БФТ.е-Архив»
СУБД	Система управления базами данных
Файл	Поименованная совокупность данных определённого размера, размещённая на внешних устройствах, рассматриваемая в процессе обработки как единое целое
ЭД	Электронный документ – представление Документа, передаваемое из информационной системы

ЭП, ЭЦП	Электронная подпись – это реквизит файла, полученный в результате криптографического преобразования
---------	---

1 Общие положения

Настоящий документ содержит сведения, необходимые для работы программиста с Системой хранения и предоставления электронных документов «БФТ.е-Архив».

1.1 Наименование программы

Полное наименование системы: Система хранения и предоставления электронных документов «БФТ.е-Архив».

Условное обозначение: Система «БФТ.е-Архив» (далее – Система, «БФТ.е-Архив»).

2 Назначение и условия применения программы

2.1 Назначение программы

Назначением Системы «БФТ.е-Архив» является обеспечение надежного и безопасного хранения, учета и поиска, передачи и получения структурированных документов, файлов и вложений документов, с использованием ЭП.

2.2 Функции, выполняемые программой

Система «БФТ.е-Архив» представляет собой совокупность функциональных модулей, реализующих функции и группы функций.

Перечень функций, реализуемых Системой:

- загрузка файлов, структурированных документов и ЭП;
- хранение архивных карточек, файлов, структурированных документов;
- просмотр файлов, структурированных документов и ЭП;
- поиск архивных карточек и файлов, структурированных документов;
- ведение и обработка ЭП;
- взаимодействие со смежными системами;
- администрирование Системы.

2.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы

2.3.1 Требования к составу и параметрам комплекса технических средств

Средства аппаратного обеспечения, необходимые для функционирования Системы, рассчитываются индивидуально под потребность решаемых задач Системой, минимальные требования представлены в таблице 1.

Таблица 1 – Сведения об аппаратном обеспечении Системы

Вид ресурса	Рекомендуемые требования
Сервер приложения	
Процессор	Не менее 16 ядер (AMD or Intel x86 64-бит, 2,4 ГГц)
ОЗУ	Не менее 32 Гб
Жесткий диск	Не менее 100 Гб, SAS (для защиты данных, рекомендуется использовать технологию RAID)
Сетевой адаптер	Не менее 1 Гбит/с
Сервер баз данных приложения	
Процессор	Не менее 8 ядер (AMD or Intel x86 64-бит, 2,4 ГГц)
ОЗУ	Не менее 16 Гб
Жесткий диск	Не менее 500 Гб, SAS (для защиты данных необходимо использовать технологию RAID)

Вид ресурса	Рекомендуемые требования
	Требования к объему жесткого диска для сервера СУБД определяются исходя из объемов хранения документов
Сетевой адаптер	Не менее 1 Гбит/с
Сервер Сервиса электронной подписи (суммарно, продуктивная среда и среда тестирования)	
Процессор	Не менее 8 ядер (AMD or Intel x86 64-бит, 2,4 ГГц)
ОЗУ	Не менее 8 Гб
Жесткий диск	Не менее 100 Гб, SAS (для защиты данных, рекомендуется использовать технологию RAID)
Сервер файлового хранилища (суммарно, продуктивная среда и среда тестирования)	
Процессор	Не менее 16 ядер (AMD or Intel x86 64-бит, 2,4 ГГц)
ОЗУ	Не менее 16 Гб
Жесткий диск	Не менее 1 Тб, SATA (для защиты данных необходимо использовать технологию RAID) Требования к объему жесткого диска для сервера файлового хранилища данных определяются исходя из: текущих объемов данных; прогноза по приросту объемов данных; требований к резервированию и отказоустойчивости
Сетевой адаптер	Не ниже 1 Гбит/с

Требования к рабочей станции программиста приведены в таблице 2.

Таблица 2 – Минимальные требования к рабочей станции программиста

Состав оборудования	Минимальные требования
Процессор	2.8 GHz Intel P4 (рекомендуемый 2.0 GHz Core2Duo и выше) или подобный
ОЗУ	4 Гб DDR
Дисковое пространство	80 Гб
Монитор с разрешением экрана	1024x768 пкс
Сетевая карта с пропускной способностью	100 Мбит/с
Сетевое подключение	не менее 1 Мбит/с, рекомендуется скорость – 10 Мбит/с

Скорость передачи данных в канале связи между Системой и пользователями должна быть не менее 1 Мбит/с, рекомендованная скорость – 10 Мбит/с.

2.3.2 Требования к общесистемному программному обеспечению

Перечень и описание Системного ПО, обеспечивающего корректную работу Системы, представлены в таблице 3.

Таблица 3 – Сведения о программном обеспечении Системы

Вид ресурса	Рекомендуемые требования
Сервер приложения	
Операционная система	Astra Linux 2.11 / 2.12 RedOS 7.x RedHat Enterprise Linux 7.x /8.x ОС Альт 8 СП
Версия Java	OpenJDK / LibericaJDK 11 (11.x)
Tomcat	Apache Tomcat 9.0
Просмотр документов	LibreOffice 6.x / 7.x
Сервер баз данных приложения	
Операционная система	Astra Linux 2.11 / 2.12 RedOS 7.x RedHat Enterprise Linux 7.x /8.x

Вид ресурса	Рекомендуемые требования
	ОС Альт 8 СП
База данных	PostgreSQL12
Сервер Сервиса электронной подписи	
Операционная система	Astra Linux 2.12 или RedOS 7.3
База данных	PostgreSQL12
Криптопровайдер	Крипто Про JCP 2.0
Версия Java	OpenJDK / LibericaJDK 8
Операционная система	Astra Linux 2.12 или RedOS 7.3
Сервер файлового хранилища	
Операционная система	Astra Linux 2.11 / 2.12 RedOS 7.x RedHat Enterprise Linux 7.x /8.x ОС Альт 8 СП

Средства, обеспечивающие функционирование Системы:

– Сервер приложения — виртуальные машины, обеспечивающие функционирование прикладного программного обеспечения;

– Сервер баз данных приложения — виртуальные машины, обеспечивающие хранение и обработку данных в БД SQL, а также функции резервного копирования и восстановления;

– Сервер Сервиса электронной подписи — виртуальные машины, предназначенные для обработки запросов от клиентского приложения на формирование и проверку ЭП переданных данных;

– Сервер балансировки нагрузки (NGINX) — виртуальные машины, предназначенные для балансировки сетевого трафика пользователей и взаимодействующих систем с Системой;

– Сервер файлового хранилища – виртуальные машины, предназначенные для хранения файлов.

Взаимодействие средств Системы осуществляется в соответствии со схемой, представленной на рисунке 1.

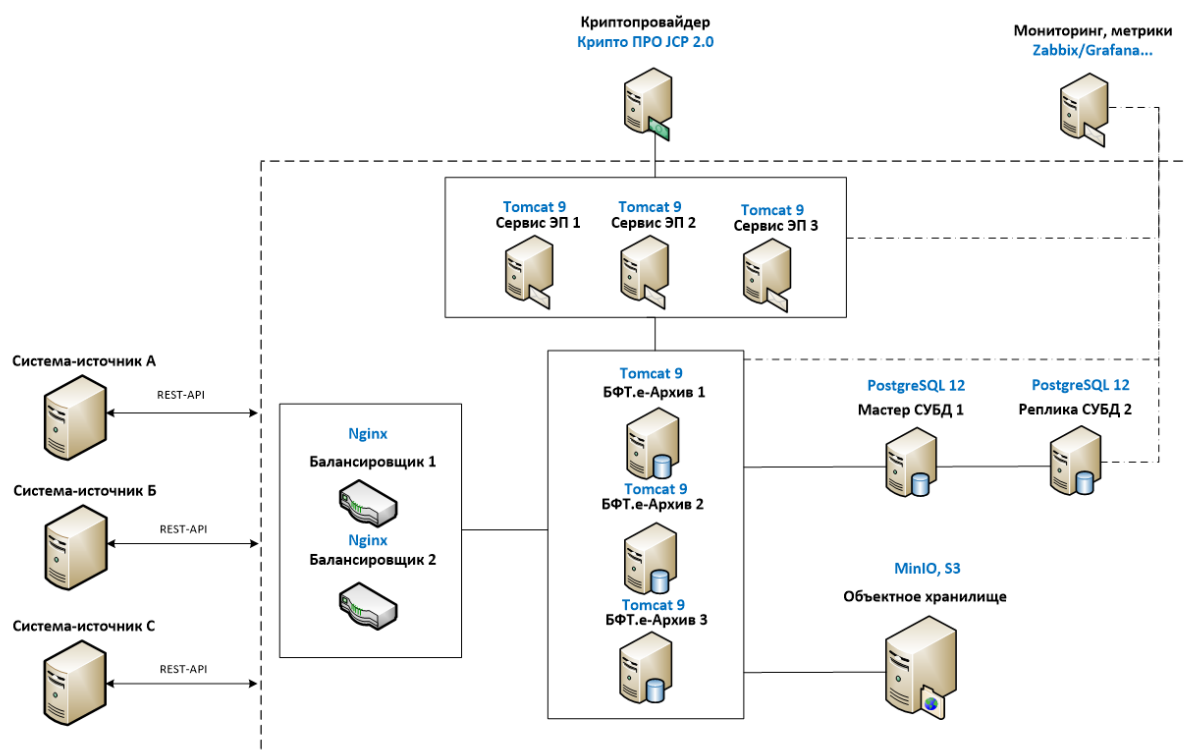


Рисунок 1 – Схема логической структуры

Для доступа к функциям Системы на компьютере программиста должно быть установлено следующее программное обеспечение:

- Операционная система;
- Веб-обозреватель (браузер) Yandex не ниже версии 22, Google Chrome не ниже версии 107.0.5304.88;
- КриптоПро ЭЦП Browser plug-in;
- Программа чтения pdf-файлов Adobe Reader или аналог из реестра российских программ для электронных вычислительных машин и баз данных Министерства коммуникаций и связи Российской Федерации;
- Пакет офисного ПО из реестра российских программ для электронных вычислительных машин и баз данных Министерства коммуникаций и связи Российской Федерации.

Перечень операционных систем, которые могут быть установлены на рабочей станции программиста:

- Microsoft Windows 7;
- Microsoft Windows 8;
- Microsoft Windows 10;
- Linux, версии которых присутствуют в реестре отечественного программного обеспечения Министерства коммуникаций и связи Российской Федерации.

2.3.3 Требования к персоналу

Программисты должны иметь навыки работы с компьютером и общим ПО (ОС, офисное ПО) в объеме навыков пользователей персональных компьютеров. В том числе обладать навыками работы с современными используемыми версиями интернет-браузеров (Google Chrome, Яндекс.Браузер, Mozilla Firefox 43.0, MS Internet Explorer 10.0, 11.0 и т.д.), а также с программными средствами MS Office (и/или другими пакетами офисного ПО из реестра российских программ для электронных вычислительных машин и баз данных Министерства коммуникаций и связи Российской Федерации).

Уровень подготовки программиста должен соответствовать категории «Программист», а сами программисты должны обладать следующими специальными навыками:

- Знание стандартных возможностей, используемых ОС семейства Linux и СУБД (PostgreSQL 12);
- Знание стандартных возможностей языка SQL;
- Знание стандартных возможностей языка Kotlin;
- Знание стандартных возможностей языка разметки XML;
- Знание текстового формата обмена данными JSON, основанный на JavaScript;
- Навыки по чтению технической документации на английском языке.

Специальная подготовка программиста должна включать получение знаний и навыков работы с комплексом технических средств и прикладного программного обеспечения «БФТ.е-Архив» в объеме, необходимом для исполнения своих должностных обязанностей.

3 Характеристика программных средств программы

3.1 Описание основных характеристик программы

3.1.1 Временные характеристики программы

В штатном режиме функционирования Система должна обеспечивать устойчивое функционирование и время реакции отклика, указанное в таблице 4. Время реакции (отклика) по некоторым сервисам может быть увеличено, если это связано с взаимодействием с внешними системами и зафиксировано в рабочей документации на Систему.

Таблица 4 – Показатели назначения Системы

Наименование показателя	Количественные характеристики
Время восстановления работоспособности программного обеспечения при сбоях и отказах	Не более 120 мин (непрерывно)
Максимальное количество одновременно работающих авторизованных пользователей	1000
Максимальное количество одновременно обрабатываемых запросов к программным интерфейсам «БФТ.е-Архив»	1000
Время ответа программных интерфейсов «БФТ.е-Архив»	Не более 5 секунд
Скачивание версии объекта (время от нажатия кнопки «Скачать» до начала скачивания)	Не более 30 сек
Сохранение объекта (время от окончания передачи данных на сервер до окончания сохранения) с одним миллионом записей	Не более 20 минут
Авторизация пользователя в «БФТ.е-Архив»	Не более 5 сек
Поиск информации по параметрам объекта	Не более 30 сек
Отбор информации (наложение фильтров) по параметрам справочника	Не более 10 сек
Выход из «БФТ.е-Архив»	Не более 10 сек
Максимальное время формирования отчета, не требующего предварительного расчета данных (сравнения данных)	Не более 30 сек
Открытие пользователем диалоговых форм и справочников	Не более 5 сек

3.1.2 Режим работы программы

Система «БФТ.е-Архив» обеспечивает работу в следующих режимах функционирования:

- Штатный режим;
- Режим технического обслуживания;
- Режим восстановления:
 - После сбоя;
 - После локального отказа;
 - После критического отказа или аварии.

Основным режимом функционирования является штатный. В штатном режиме должна обеспечиваться доступность всех функций «БФТ.е-Архив».

Режим технического обслуживания предназначен для проведения работ по обслуживанию «БФТ.е-Архив».

Перевод «БФТ.е-Архив» из штатного режима в режим технического обслуживания и обратно должен производиться эксплуатирующим персоналом.

В случае возникновения сбоев в работе, локальных отказов, критических отказов или аварий «БФТ.е-Архив» должна переводиться в режим восстановления до полного восстановления работоспособности реализуемых функций.

Перевод «БФТ.е-Архив» из штатного режима в режим восстановления и обратно должен производиться эксплуатирующим персоналом с использованием средств автоматизации.

В штатном режиме функционирования Система обеспечивает работу пользователей в режиме – 24 часа в сутки, 7 дней в неделю (24×7).

3.1.3 Средства контроля правильности выполнения программы

Контроль правильности выполнения Системы осуществляется визуально, путем выполнения следующих действий:

- Ввести адрес сервера Системы в адресной строке интернет обозревателя.
- В открывшейся форме авторизации указывается логин и пароль пользователя в соответствующих полях.
- Нажать кнопку «Войти».

В случае ввода верного логина и правильного пароля, принадлежащих пользователю по его учетной записи, зарегистрированной в Системе, осуществится вход в главное окно системы, в ином случае проверяется актуальность (статус) учетной записи. После входа в главное окно системы необходимо проверить работоспособность Программы (кнопки меню, выполнение операций и т.д.). В работоспособном состоянии Системы на экране отображается главное окно клиентского приложения без отображения пользователю сообщений о сбое в работе.

3.1.4 Средства контроля самовосстанавливаемости программы

Самовосстанавливаемость Системы обеспечивается средствами перезапуска или резервного копирования БД. Перезапуск Системы осуществляется через остановку и последующий запуск. Описание выполняемых действий приведено в документе «Руководство системного программиста». Контроль восстановления Программы возможно проводить по работоспособности приложения или средствами штатного мониторинга.

4 Обращение к программе

4.1 Запуск процедур для интеграции с внешними системами

Для интеграции с внешними информационными системами используется единый унифицированный API в REST-архитектуре с использованием брокера сообщений Active MQ или Apache Kafka.

При выпуске новых релизов обеспечивается обратная совместимость API за счет поддержки ранее опубликованных контрактов. Механизм версионирования поддерживается за счет заведения группы URL под опубликованную версию.

Основной перечень методов API для взаимодействия с внешними системами включает следующие методы:

- Логирование (авторизация);
- Сохранение документа;
- Сохранение вложений документа;
- Сохранение подписей для вложения документа;
- Получение документа;
- Получение вложения;
- Получение подписи вложения;
- Удаление вложений из последней версии документа;
- Удаление подписей вложения.

Детальное описание указанных методов API представлено в приложении (Приложение А). Полный перечень методов размещен по адресу: [ip:8082/app/documentation/das/v1.0#/.](ip:8082/app/documentation/das/v1.0#/)

4.2 Создание скриптов

4.2.1 Создание скриптов для планирования заданий в программе

Создание скриптов возможно в «Планировщике заданий» Системы, для этого необходимо выполнить следующие действия:

- Войти в меню в раздел «Администрирование»;
- Выбрать «Планировщик заданий»;
- Создать новое задание с помощью кнопки **+** «Добавить»;
- Заполнить незаполненные обязательные поля: «Наименование» и другие формы «Планирование задания» (Рисунок 2).

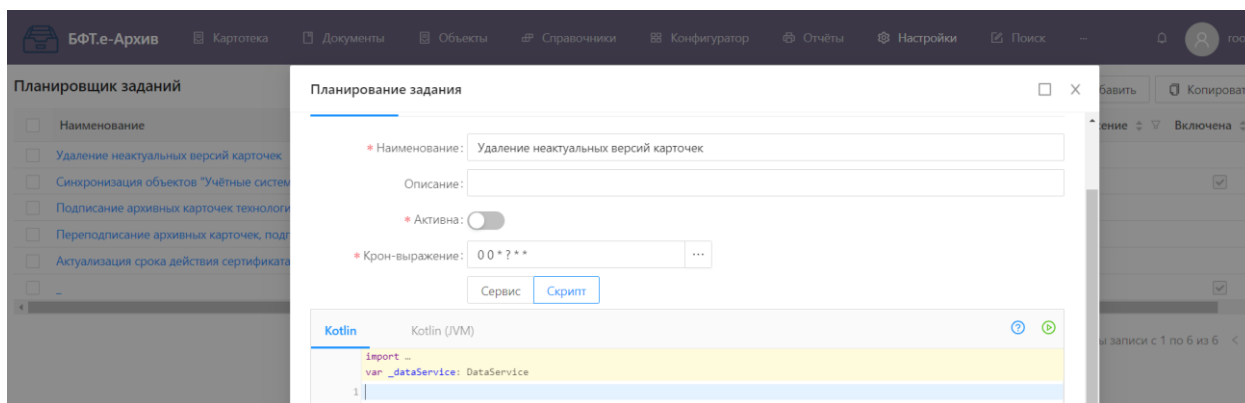


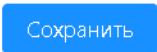
Рисунок 2 – Создание скрипта в задании «Планировщика заданий»

Описание полей формы записи приведено в таблице 5.

Таблица 5 – Поля формы записи справочника «Планировщик задач»

Наименование	Описание	Способ заполнения
Активна	Если опция включена, задание планировщика выполняется с указанным периодом запуска	Ручной ввод. Параметр имеет два состояния: «включен» и «выключен». По умолчанию параметр «выключен»
Крон выражение	Запись, определяющая периодичность выполнения задания. Запись вносится в крон-формате	Автоматически из вызываемого редактора Крон-выражений
Задание	Имя исполняемого файла	Выбор из справочника «Задачи для планирования»
Аргумент	Правило определения переменной для Задачи. Отображается только если для задачи установлена необходимость задания одной или нескольких переменных. Количество полей аргумента зависит от количества переменных выбранной Задачи	Ручной ввод. Задается скриптовым языком

В форме создания скрипта доступны следующие действия:

 Сохранить

– сохраняет запись с закрытием формы создания скрипта.

 Отменить

– закрывает форму создания скрипта без сохранения изменений.



– закрывает форму создания скрипта без сохранения изменений.

Запуск заданий планировщика, содержащих Крон-выражение, выполняется автоматически, если признак «Активно» включен.

Примечание – при внесении изменений в скрипты, установленные для правил в

Системе, следует нажать на кнопку  для того, чтобы скрипт был применен.

После проведения требуемых настроек необходимо выполнить последовательно действия «Сохранить» и «Применить» по нажатию на соответствующие кнопки.

4.2.2 Примеры скриптов в Kotlin для Планировщика заданий

Пример обновления данных почтового адреса и телефона для документов:

```
_dataService.inNewTransaction{
    val user = _dataService.find_ (on ("UserAccount").filter {f("name") eq "test"})
        user["email"] = "updated123@mail.ru"
        user["phone"] = "+7 111 222-33-44"
}
```

Пример изменения наименования документа путем добавления слова «проверка»:

```
val docs = _dataService.findAll (on ("Document").full().filter {f("srcSystem.code") eq "AHD"})
_dataService.inNewTransaction{
    docs.forEach { doc->
        val name = doc["name"] as String
        doc["name"] = "$name проверка"
    }
}
```

Пример очищения последних изменений наименований документов:

```
val docs = _dataService.findAll (on ("Document").full().filter {f("srcSystem.code") eq "AHD"})
_dataService.inNewTransaction{
    docs.forEach { doc->
        val name = doc["name"] as String
        val name2 = name.substringBefore("проверка")
        doc["name"] = "$name2".trimIndent()
    }
}
```

4.2.3 Используемые регулярные выражения в скрипте

Задание переменной для регулярного выражения производится:

val <Имя переменной> – присвоение имени переменной для правила

Шаблон регулярного выражения создается с помощью метода toRegex():

val <Имя переменной> = "^<регулярное выражение>\$".toRegex()

Задание регулярного выражения приведено в таблице 6.

Таблица 6 – Задание регулярного выражения

Регулярное выражение	Обозначение
^	Начало строки (поля). Если используется внутри символьного класса, отвечает за отрицание
.	Любой символ, кроме перевода строки \n
[A-ZА-Я]	Диапазон значений, в данном случае, больших букв. Например, диапазон [0-9a-z_] позволит задать в поле любую цифру, строчную букву или символ подчеркивания
[a-я a-z]	Диапазон значений, в данном случае, малых букв

?	Одно или ноль вхождений символа, установленного перед символом «?». Например «a?» позволит установить одно или ноль вхождений символа «a»
+	Одно или более вхождений, эквивалентно {1,}
\$	Конец строки
\	Экранирование символа. Например, при необходимости задать символ «.» используется \. Специальные символы должны быть экранированы дважды \\ \\
()	Группировка
[]	Класс символов
[...]	Любой из символов, указанных в скобках
[^...]	Любой из символов, не указанных в скобках
*	Соответствует предыдущему символу, повторенному 0 или более раз, эквивалентно {0,}
	Оператор чередования (альтернатива), соответствует либо подвыражению слева, либо подвыражению справа
\s	Соответствует символу пробела
\w	Соответствует символу слова; эквивалентно[a-zA-Z_0-9]

Задание переменной для логического выражения производится:

<Имя переменной>.containsMatchIn(_value as String) – логическое выражение для срабатывания правила валидации. Метод containsMatchIn указывает, может ли регулярное выражение найти хотя бы одно совпадение в указанном вводе, выполняет поиск соответствия шаблону в любом месте всей входной строки. Если во входной строке есть хотя бы одно совпадение, возвращается истина (true), в противном случае – ложь (false).

5 Входные и выходные данные

5.1 Характер и организация входных данных

Исходные данные в Систему поступают на основании запросов в формате JSON Schema посредством REST.API и с возможностью загрузки файлов вложений и ЭП из смежной системы. Для этого необходимо выполнить функции, осуществляемые компонентой взаимодействия со смежными системами, представленными в таблице (Таблица 7).

Таблица 7 – Перечень функций компоненты взаимодействия со смежными системами

№	Функция	Соответствие методам API	Режим доступа	Передача
1	Поддержка REST-архитектуры при взаимодействии со смежными системами	Метод логирования (авторизации) и другие поддерживаемые методы	Круглосуточно, кроме периода регламентных работ	Интеграционный сервис
2	На основании запросов из смежных систем, создание/изменение документов с загрузкой файлов документов и ЭП	Методы: сохранение карточки документа; сохранение вложений документа; сохранение подписей для вложения документа	Круглосуточно, кроме периода регламентных работ	Интеграционный сервис
3	На основании запросов из смежных систем выгрузка документов, файлов и ЭП	Методы: получение документа; получение вложения; получение подписи вложения	Круглосуточно, кроме периода регламентных работ	Интеграционный сервис
4	На основании запросов из смежной системы, удаление файлов документов из последней версии и ЭП	Методы: удаление вложений из последней версии документа; удаление подписи из вложения	Круглосуточно, кроме периода регламентных работ	Интеграционный сервис

5.1.1 Описание входных данных, получаемых с использованием API

Описание входных данных, получаемых с использованием API, приведено в Приложении А.

5.2 Характер и организация выходных данных

Описание выходных данных аналогично описанию входных данных и приведено в разделе 5.1 настоящего документа.

6 Сообщения

При работе с Системой могут выводиться сообщения об ошибках, требующие выполнения определенных действий.

Описание ошибок или действий выводится с сообщением.

Например, при вводе неправильного логина и/или пароля выводится сообщение, представленное на рисунке 3.

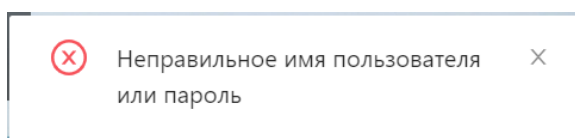


Рисунок 3 – Сообщение Системы о неверно введенном логине или пароле

Пример сообщения об ожидании заполнения символов при компиляции скрипта представлен на рисунке 4.

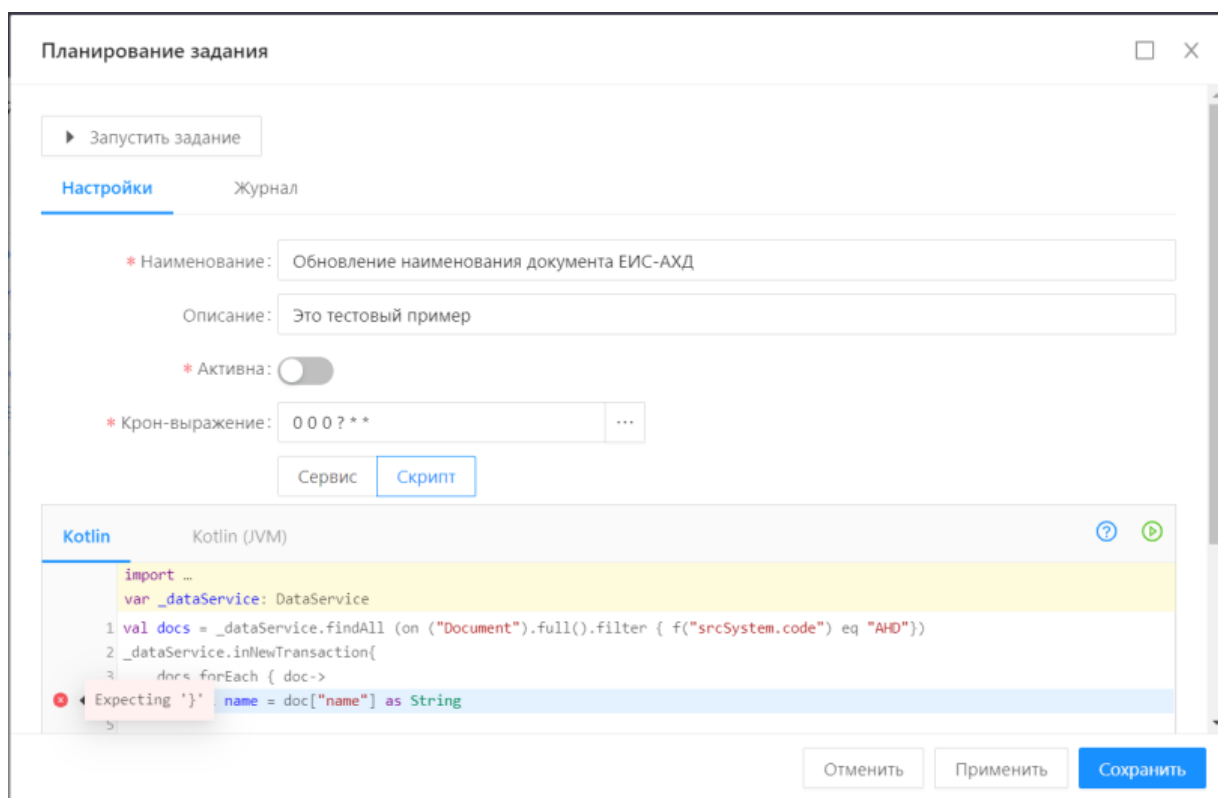


Рисунок 4 – Сообщение Системы о необходимости выполнить действие

Приложение А

Описание программного интерфейса приложения (API)

Описание методов сервиса «БФТ.е.Архив»

Путь к методам: http://server:port/app/

Полный перечень методов по адресу: ip:8082/app/documentation/das/v1.0#/

А.1 Получение токена логина

Метод: POST

Путь: /api/auth/login

Параметры: No parameters

Тело запроса метода получения токена логина в Системе приведено в таблице А.1.

Таблица А.1 – Тело запроса метода получения токена логина в Системе

Наименование	Обязательный	Описание/Пример запроса
Body (body)	+	Имя пользователя и пароль. <pre>{ "username": "user", "password": "pass" }</pre>

Тело ответа из Системы на запрос метода получения токена логина приведено в таблице А.2.

Таблица А.2 – Тело ответа из Системы на запрос метода получения токена логина

Код ответа	Описание/Пример ответа
200	Успешное выполнение. <pre>{ "accessToken": "eyJhbGciOiJIUzUxMiJ9.eyJhdXRocyI6IiJJPTEVfc3VwZXJvc2VyIiwic3ViIjoicm9vdCIsImIhdCI6MTUzODAzMjIyMSwiZXhwIjozNTM4MDE4fQ.cOwyPsDw9_yYwjl69eojagt0zSJi3vT3eb0yd5AiWVkw41bj06YXRrheI38P4V-f7L2R46MAB_QenZt-1qpwKg", "tokenType": "Bearer", "mustChangePassword": false, "onlySSOEnabled": false, "useAlternativeSso": false, "timezone": "Europe/Moscow", "dbType": "PostgreSQL" }</pre>
400	Invalid status value

А.2 Сохранение документа

Метод: POST

Путь: /app/api/v1.0/structuredDocument/store

Параметры: No parameters

Тело запроса метода сохранения нового документа в Системе приведено в таблице А.3.

Таблица А.3 – Тело запроса метода сохранения нового документа в Системе

Наименование	Обязательный	Описание/Пример запроса
Body (body)	+	Сохранение нового документа

Наименование	Обязательный	Описание/Пример запроса
		<pre> { "name": "Документ для теста", "organisation": { "name": "ПАО \"Газпром\"", "inn": "694332675485", "ogrn": "4621770627719", "kpp": "349465342" }, "docType": { "name": "Договор", "code": "MODULE_TEST.test_doc" }, "srcSystem": "SAUMI", "docNumber": "43095572200948297191", "docSumm": "7DD987F846400079F4B03C058365A4869047B4A0", "docDate": "2020-03-12T15:44:37", "dtCreated": "2020-03-12T15:44:37", "dtEnd": "2020-03-12T15:44:37", "dtChanged": "2020-03-12T15:44:37", "extId": "999584", "attachs": [{ "extId": "attach-0", "fileName": "file.txt", "attachFileType": "txt", "attachType": "attach", "attachCategory": "категория 1", "author": "Пушкин", "checksum": "7DD987F846400079F4B03C058365A4869047B4A0", "checksumAlg": "MD5", "file": "string", "paramName": "attach1", "signatures": [{ "file": "string", "extId": "1234444", "filename": "1234444", "certificate": "string", "format": "cms", "signdate": "2003-11-07T00:00:00", "paramName": "signature1" }] }] }, "docBody": { "name": "Документ - 34546457", "some_test_number": 0, "some_test_date": "2020-03-12T15:44:37" } } </pre>

Тело ответа на запрос метода сохранения нового документа в Системе приведено в таблице А.4.

Таблица А.4 – Тело ответа на запрос метода сохранения нового документа в Системе

Код ответа	Описание/Пример ответа
200	Успешное выполнение. {

	<pre> "id": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "name": "Документ 2003-5", "docNumber": "43095572200948297191", "docSumm": "7DD987F846400079F4B03C058365A4869047B4A0", "dtChanged": "2003-11-07T00:00:00", "docDate": "2003-11-07T00:00:00", "dtCreated": "2003-11-07T00:00:00", "dtEnd": "2004-09-03T00:00:00", "extId": "999584", "version": 0, "docType": { "name": "Договор", "code": "MODULE_TEST.test_doc" }, "nomenclature": { "name": "Документ договора аренды муниципального помещения", "code": "01.202" }, "srcSystem": { "name": "SAUMI", "code": "SAUMI" }, "organisation": { "name": "ПАО \"Газпром\"", "inn": "694332675485", "ogrn": "4621770627719", "kpp": "349465342" }, "attachs": [{ "id": "86c59182-c513-460e-9df0-7e0cca2a748b", "extId": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "fileName": "file.txt", "signatures": [{ "id": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "extId": "123123", "fileName": "attach1_fileName_0.sig" }] }] </pre>
400	Invalid status value
401	Необходимо установить токен авторизации

А.3 Сохранение вложений документа

Метод: POST

Путь: /app/api/v1.0/structuredDocument/storeAttachmentsById

Параметры метода сохранения вложений документа в Системе приведены в таблице А.5.

Таблица А.5 – Параметры метода сохранения вложений документа в Системе

Наименование	Обязательный	Описание/Пример запроса
typeCode (string [query])	+	Код типа документа
documentId (string [query])	+	Внутренний идентификатор документа

Тело ответа на запрос метода сохранения вложений документа в Системе приведено

в таблице А.6.

Таблица А.6 – Тело ответа на запрос метода сохранения вложений документа в Системе

Код ответа	Описание/Пример ответа
200	Успешное выполнение. <pre>{ "id": "86c59182-c513-460e-9df0-7e0cca2a748b", "extId": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "fileName": "file.txt", "signatures": [{ "id": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "extId": "123123", "fileName": "attach1_fileName_0.sig" }] }</pre>
400	Invalid status value
401	Необходимо установить токен авторизации

А.4 Сохранение подписей для вложения документа

Метод: POST

Путь: /app/api/v1.0/structuredDocument/storeAttachmentSignatures

Параметры метода сохранения подписей для вложения документа в Системе приведены в таблице А.7.

Таблица А.7 – Параметры метода сохранения подписей для вложения документа в Системе

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа
attachmentId (string [query])	+	Внутренний идентификатор вложения

Тело ответа на запрос метода сохранения подписей для вложения документа в Системе приведено в таблице А.8.

Таблица А.8 – Тело ответа на запрос метода сохранения подписей для вложения документа в Системе

Код ответа	Описание/Пример ответа
200	Успешное выполнение. <pre>{ "newAttachmentId": "86c59182-c513-460e-9df0-7e0cca2a748b", "newSignatures": [{ "id": "eed7723e-d21a-4260-a0e1-b58bc4a9947a", "extId": "123123", "fileName": "attach1_fileName_0.sig" }] }</pre>
400	Invalid status value
401	Необходимо установить токен авторизации

А.5 Получение документа

Метод: GET

Путь: /app/api/v1.0/structuredDocument/readDocument

Параметры метода получения документа из Системы приведены в таблице А.9.

Таблица А.9 – Параметры метода получения документа из Системы

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа
withData (boolean [query])	+	Загружать ли вложения в дополнении к карточке

Тело ответа на запрос метода получения документа из Системы приведено в таблице А.10.

Таблица А.10 – Тело ответа на запрос метода получения документа из Системы

Код ответа	Описание/Пример ответа
200	zip-файл с документом
401	Undocumented
404	Error или Not Found

А.6 Получение вложения

Метод: GET

Путь: /app/api/v1.0/structuredDocument/readDocumentAttachment

Параметры метода получения вложения из Системы приведены в таблице А.11.

Таблица А.11 – Параметры метода получения вложения из Системы

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа
attachmentId (string [query])	+	Внутренний идентификатор вложения

Тело ответа на запрос метода получения вложения из Системы приведено в таблице А.12.

Таблица А.12 – Тело ответа на запрос метода получения вложения из Системы

Код ответа	Описание/Пример ответа
200	zip-файл с документом
401	Undocumented
404	Error или Not Found

А.7 Получение подписи вложения

Метод: GET

Путь: /app/api/v1.0/structuredDocument/readDocumentAttachment

Параметры метода получения подписи вложения из Системы приведены в таблице А.13.

Таблица А.13 – Параметры метода получения подписи вложения из Системы

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа
attachmentId (string [query])	+	Внутренний идентификатор вложения

Наименование	Обязательный	Описание/Пример запроса
signatureId (string [query])	+	Внутренний идентификатор подписи

Тело ответа на запрос метода получения подписи вложения из Системы приведено в таблице А.14.

Таблица А.14 – Тело ответа на запрос метода получения подписи вложения из Системы

Код ответа	Описание/Пример ответа
200	zip-файл с документом
401	Undocumented
404	Error или Not Found

А.8 Удаление вложений из последней версии документа

Метод: DELETE

Путь: /app/api/v1.0/structuredDocument/deleteAttachments

Параметры метода удаления вложений из последней версии документа в Системе приведены в таблице А.15.

Таблица А.15 – Параметры метода удаления вложений из последней версии документа в Системе

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа
attachmentIds (array[string] (query))	+	Массив внутренних идентификаторов вложений

Тело ответа на запрос метода удаления вложений из последней версии документа в Системе приведено в таблице А.16.

Таблица А.16 – Тело ответа на запрос метода удаления вложений из последней версии документа в Системе

Код ответа	Описание/Пример ответа
200	Ответ - идентификатор новой версии карточки документа <pre>{ "newDocumentId": "247bf78b-35c7-445a-99ff-a48b30165493", "newAttachmentId": "247bf78b-35c7-445a-99ff-a48b30165493" }</pre>
401	Undocumented
404	Error или Not Found

А.9 Удаление подписей вложения

Метод: DELETE

Путь: /app/api/v1.0/structuredDocument/deleteAttachmentSignatures

Параметры метода удаления подписей вложения в Системе приведены в таблице А.17.

Таблица А.17 – Параметры метода удаления подписей вложения в Системе

Наименование	Обязательный	Описание/Пример запроса
documentId (string [query])	+	Внутренний идентификатор документа
typeCode (string [query])	+	Код типа документа

Наименование	Обязательный	Описание/Пример запроса
attachmentId (string [query])	+	Внутренний идентификатор вложения
signatureIds (string [query])	+	Массив внутренних идентификаторов подписей

Тело ответа на запрос метода удаления подписей вложения в Системе приведено в таблице А.18.

Таблица А.18 – Тело ответа на запрос метода удаления подписей вложения в Системе

Код ответа	Описание/Пример ответа
200	Ответ - идентификатор новой версии карточки документа и идентификатор вложения <pre>{ "newDocumentId": "247bf78b-35c7-445a-99ff-a48b30165493" }</pre>
401	Undocumented
404	Error или Not Found